

HotDocs Server 11.2

PRODUCT OVERVIEW

Copyright

© 1996-2019 AbacusNext.

All rights reserved. No part of this product may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language in any form by any means without the express written permission of HotDocs Limited. (“HotDocs”)

Warranty Information

HotDocs makes no representations or warranties with respect to the contents or use of this product and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Information in this document is subject to change without notice and does not represent a commitment on the part of HotDocs.

Government Use

Use, duplication, or disclosure by the Federal Government is subject to restrictions as set forth in FAR clauses 52.227--14, “Rights in Data--General”; 52.227--19, “Commercial Computer Software--Restricted Rights”; and subdivision (c)(1)(ii) of the Rights in Technical Data and Computer Software clause DFAR 252.227--7013; and the limitations set forth in the standard commercial license agreement for this software. Unpublished rights are reserved under the copyright laws of the United States.

Trademark Information

HotDocs is a registered trademark of HotDocs Limited. Other product names may be trademarks or registered trademarks of their respective companies.

Sample Templates and Forms

This product includes sample templates and forms only. Using them may have significant legal implications in some situations, and these implications vary by state and depending on the subject matter. Before using these templates and forms or adapting them for your business, you should consult with a lawyer and financial advisor.

Table of Contents

About HotDocs	5
What is HotDocs Server?	7
HotDocs Server Architecture	9
Integrating with HotDocs Server?	10
What is a HotDocs Server host application?	13
What is HotDocs Developer?	16
System Requirements	17

About HotDocs

What is HotDocs document assembly software?

HotDocs is an award-winning software application that significantly reduces the time spent generating customized documents, such as contracts, sales proposals, government and court forms, legal documents, loan applications, and medical forms.

Using HotDocs, you can transform any word processor file or form into an interactive template by replacing the changeable text with HotDocs variables. Then, the next time you want to generate a completed document, just assemble the interactive template you've created. As you do this, you will be prompted for the information needed in the document and that information will be merged into the document.

When using HotDocs, you can perfect a template, minimizing the “human error” factor that repetitive typing introduces. Additionally, you can automate your templates so that verb tenses, gender references, dates, numbers, calculations, and more are updated automatically as users enter information. Custom interview questions and help resources guide you or your users through the interview.

Virtually any document in your workflow can be converted to a HotDocs template. HotDocs templates then become your gold standard—experienced colleagues can share their knowledge, and new colleagues come up to speed faster. Using HotDocs, you can compile an invaluable repository of special language, unique clauses or terms, correspondence, and more.

Templates and Component Files

HotDocs supports the following three types of templates:

Template Type	Description
	<p>A text template is created and automated in a word processor, such as Microsoft Word or WordPerfect. You can modify the underlying text of a text template both as you automate the template and as you view the assembled document in the word processor. Text templates have a .DOCX, .RTF, .DOT, .WPT or .TTX file name extension.</p> <p>--- Caution ---</p> <p>Microsoft Word .DOT templates may not be used with HotDocs Server.</p>
	<p>A form template is based on static graphical or text content, such as a PDF document. Variable fields are placed on top of the static content where answers will be overlaid during assembly, but the underlying static text and formatting does not change. Form templates have a .HFT or .HPT file name extension</p>
Interview Templates	<p>An interview template gathers specific information (such as court, attorney, or client information) and saves the answers for use in assembling other documents. Unlike text and form templates, assembly of an interview template does not generate a document directly—it generates only an answer file. However, an interview template can trigger subsequent assembly of additional documents using the ASSEMBLE instruction. Interview templates have a .CMP file name extension.</p>

Tip: Interview templates only have a component file; they do not have a separate template file. Templates used with HotDocs Server also require an additional file: the template manifest (.manifest.xml). Like a component file, this file also shares the same base file name as the template except for the file name extension. It is created when you publish a template for use with HotDocs Server.

When you create a new template file, HotDocs automatically creates a companion file called the component file. The component file contains information about variables and other components used in the template. The component file has the same base file name as the template file, but with a .CMP file name extension. Both the template file and the component file are necessary for a template to work. Whenever you copy a HotDocs template—for example, to share a template with another user—you must be sure to copy both the template file and the component file.

The component file works automatically in the background—as you create various components in the template, they are automatically stored in the component file. Normally each template uses its own component file, but you can make two or more templates share one component file.

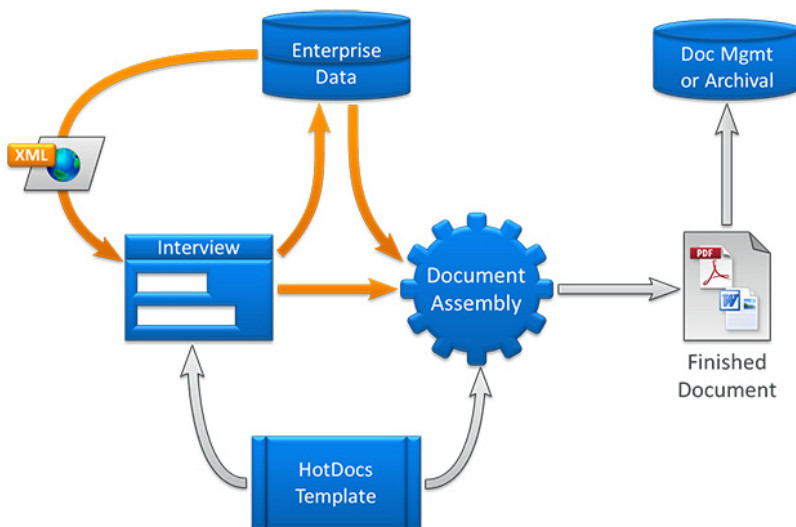
--- **Note** ---

Versions of HotDocs Developer prior to 11 uploaded JavaScript (.JS) and Variable Collection (.HVC) files when publishing content for use with HotDocs Server. These files are no longer of concern when working with HotDocs Server version 11 or later.

What is HotDocs Server?

HotDocs Server is the server-based version of HotDocs that allows HotDocs interviews (data-gathering sessions) to be displayed by a standard Web browser and documents to be generated (assembled) on the web server, without requiring any special software to be installed on an end-user's computer. Generally speaking, when a user goes to your Web site and initiates assembly of a document, a request is made to the HotDocs Server engine, which is running on the server. HotDocs Server then sends an interview back to the user's browser, where it appears as part of a Web page. Once the required information has been entered, the user clicks a button that posts the answers (in XML format) back to HotDocs Server, where they can be merged into assembled documents.

The following diagram shows the assembly process:



The benefit of creating a Web application using HotDocs Server is that end users do not need HotDocs installed on their computers, since the interview is presented in a Web browser and document assembly happens on the server. For example, your human resources department may integrate its forms into its intranet site using HotDocs Server. Employees would then go to the intranet site, select which form to fill out, provide the required information, and then submit the form to the appropriate human resources representative—all from within a Web browser. Depending on project requirements, employees may also print copies of assembled documents, save copies of assembled documents to their hard drives, or (if the intranet site facilitates it) simply store assembled documents or answer files directly in the intranet app.

HotDocs Server may also be used to generate a document without an interactive interview. Likewise, HotDocs browser interviews may be used to gather information for storage or later use without immediate assembly of any documents at all.

HotDocs Server vs. Desktop HotDocs

The HotDocs family of products includes both desktop- and server-based products. The primary desktop-based product, HotDocs Developer, is used for creating HotDocs templates. HotDocs Server, on the other hand, is used only for running templates; it cannot be used to create them. Another difference between Server and desktop HotDocs is the user interface: desktop editions of HotDocs include word processor and other integrations for creating templates, and the “HotDocs Library” for managing templates, answer files, and more. The user interface presented directly by HotDocs Server, at least from an end user's perspective, is limited to displaying interviews in the Web browser. All other interaction with the user is accomplished by a custom host application, which you must build and maintain. The advantage with this approach is that you can expose your users to only as much user interface as they need to perform their document assembly tasks.

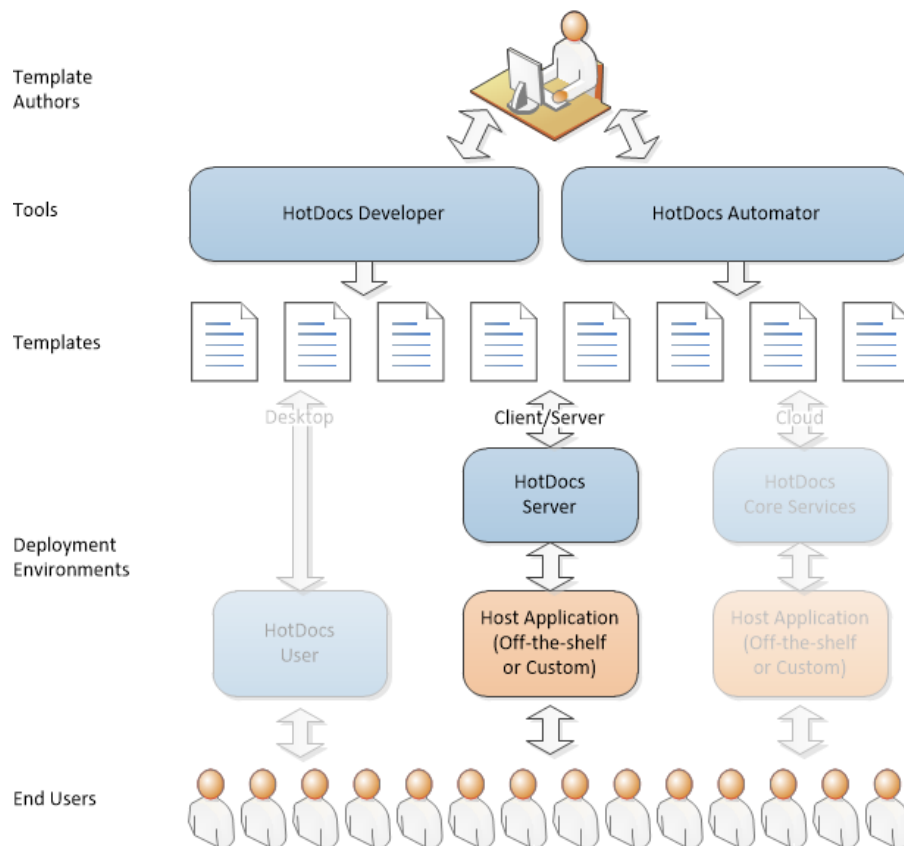
Despite their differences, desktop HotDocs still plays an important role in relation to HotDocs Server: it is the tool used to create (and help test) all content (templates) that will eventually be deployed in HotDocs Server.

HotDocs Server Architecture

Implementing HotDocs in your organization involves the integration of several modules or applications. At a minimum, a HotDocs Server deployment consists of the following:

- HotDocs Developer and Automator (for template development)
- HotDocs Server (for template deployment)
- Host Application (for making templates available to end users)

Each module plays a unique role in providing a complete solution to your organization's document automation and data collection needs.



Integrating with HotDocs Server?

At the center of HotDocs Server is a Windows operating system service that runs in the background and performs various tasks, such as assembling documents, producing HTML interviews, and handling answer files. As an operating system service, HotDocs Server starts when Windows is started and runs in its own process. This means that HotDocs Server is always available, even when no users are logged on to the system. Also, because HotDocs Server runs in its own process rather than running in a thread belonging to another application, security for HotDocs threads is not dictated by the client process.

The core HotDocs Server service does not have a user interface. Instead, it provides the following APIs through which external applications (with their own user interfaces) can integrate with HotDocs Server and instruct it to perform various tasks:

COM API	The HotDocs Server COM API is the fundamental interface for integrating with HotDocs Server from a host application running on the same machine. It consists of a set of COM objects which allow you to retrieve interviews suitable for display in a Web browser, pre-populate those interviews with existing data, assemble documents, and perform many other tasks relevant to Web-based document assembly.
.NET API	The HotDocs Server .NET API is a set of native .NET classes built on top of the COM API, which you can use to integrate HotDocs Server with your own custom .NET host application. It provides essentially the same set of features as the COM API, but also incorporates some convenient .NET features such as IEnumerable collections and IDisposable implementations.
Web Service API	The web services API installs separately from HotDocs Server itself. It is a web service, hosted in IIS, which exposes fundamental HotDocs Server features to HTTP callers via the SOAP protocol. Organizations who need a host application to run on a different machine (or a different platform) than HotDocs Server can choose to install and use the web services API, and then design their host application to access HotDocs via this SOAP interface. Source code & documentation for the web services API available separately.

Note

By default, the HotDocs Server operating system service runs under the SYSTEM user account. This is a special account with full privileges to read and write files on the local file system, but it cannot access files on a network. If you would like HotDocs Server to run under a different user account, consult your operating system documentation. (If you do configure the HotDocs Server service to run under a different user account, you must take care to give that account Read, Write, and Modify permissions to the temporary files folder, as well as to the template, answer, and assembled document files folders of your host application.)

Browser Interviews

One of the services provided by HotDocs Server is the generation of a **browser-based interview** where questions are presented to end users and answers are gathered. The following picture shows an example of a browser interview:

The screenshot shows a web interface for an 'Employment Agreement' interview. The page has a blue header with 'Sample Portal for HotDocs® Server' and 'Site Powered By HotDocs® Server'. A navigation bar includes '> Home' and a 'Help' button. On the left, an 'Interview Outline' sidebar lists sections: Employee Information, Agreement Information (selected), Job Information, and Salary and Benefits. The main content area, titled 'Agreement Information', contains the following fields:

- Agreement Date:** 5/4/2010
- Company Representative:** A list with radio buttons for Stephanie Hanson (selected), Ed Walters, and Kim Schuster.
- Signature Date (Leave blank if unknown):** An empty date field.

Navigation buttons at the bottom include 'First', 'Previous', 'Next', 'Last', and 'Finish'. A summary box at the bottom states: 'Stephanie Hanson signs for editorial employees, Ed Walters for marketing, and Kim Schuster for production'.

Browser interviews come in two varieties: JavaScript + HTML (usually referred to as simply “JavaScript”) and Microsoft Silverlight. Silverlight interviews allow for high-performance execution and can be preferable when dealing with very large or complex interviews, while JavaScript interviews are able to work across a broader spectrum of standards-compliant browsers and devices. Both types of interviews support the same base feature set.

A number of factors affect the appearance of browser-based interviews. First of all, the host application determines the layout of the Web page on which the interview is displayed. HotDocs Server provides methods that can be called to produce an interview, which is then embedded in the host application’s Web page. The interview may be displayed by itself on a page, or it may be included on a page with other host application content -- graphics, links, menus, etc.

Style sheets and the format of the interview also affect its appearance. In addition to the core differences in appearance associated with interviews in JavaScript or Silverlight format, each of these interview formats uses a style sheet that you can customize to make the interview blend in with your host application. For example, if your host application uses a common set of colors and fonts, you can modify the style sheet to use the same colors and fonts in the interview, thus giving the interview a uniform appearance with the rest of the Web site.

Template developers that create templates for use with HotDocs Server can also greatly influence the appearance of interviews. For example, they can group related variables in dialogs, place variables side-by-side in dialogs, add additional text or prompts to explain questions, or include resource links to help users complete the interview.

Browser-based interviews expose an API allowing host applications to customize the user interface and enhance integration in other ways:

API	Description
	The HotDocs Browser Interview JavaScript API includes a number of JavaScript functions, callbacks and global variables you can use on the client (browser) side of your host application to interact with the browser-based interview.

Scaling

HotDocs Server is an efficient and high-performance system for mass-producing documents. However, if your document generation needs exceed the capacity of a single server box, HotDocs Server can be load-balanced as long as certain guidelines are followed in the development of your host application. HotDocs is highly suited to being deployed in load-balanced clusters behind industry-standard load balancing. Please see the HotDocs Web Services documentation for more details on using HotDocs in a load-balanced environment.

What is a HotDocs Server host application?

A HotDocs Server host application is any web application that utilizes HotDocs Server features on the back end to accomplish tasks related to document generation. Some host applications are focused primarily on document generation, but this is not always the case. Many third-party custom host applications focus primarily on other areas of functionality (for example, enterprise-specific workflows) and use HotDocs whenever documents need to be generated. Either way, the designation of "host application" generally refers to the application that integrates with (or "hosts") HotDocs. If you think of HotDocs as an engine, the host application is analogous to the automobile that makes the engine's power available to an end user.

Who writes and maintains a host application? A custom-built solution can be developed by someone in your organization or an outside consultant. It is also possible to acquire (and possibly adapt) an out-of-the-box, general-purpose host application that is developed and maintained by HotDocs Ltd. or another other third party. Although host applications are frequently ASP.NET Web applications, if you choose integrate with HotDocs Server via its web services interface, the host application can in fact be almost any type of application on any platform (Java, PHP, etc.).

Division of Labour

There are several features typically exposed by a HotDocs host application. Some of these features can be accomplished entirely by HotDocs Server; some are provided by the host application. This leads to a flexible architecture that shields end users from much direct interaction with HotDocs and imposes very few requirements on the end user experience. Depending on the design and functionality of your host application, end users may not even be aware HotDocs Server is in use, but they will naturally benefit from the powerful document generation capabilities HotDocs Server brings to the table.

Following are the primary tasks that host applications are concerned with:

Feature	HotDocs Server's Role	Host Application's Role
	HotDocs Server requires that templates (and all their dependent files) be accessible by direct file path (either local paths or UNC network paths are supported).	The host application is responsible for managing the upload and storage of templates and the selection of which templates to use when.
Manipulate Answers	Data is supplied to interviews or document assemblies using an XML-based Answer Collection (also known as an Answer File). HotDocs Server's COM and .NET APIs provide Windows-based host applications with a means of easily producing, inspecting, and manipulating Answer Collections.	Host applications can provide custom data to pre-populate interviews, or even assemble documents without presenting an interview at all. Being conversant in HotDocs Answer Collections allows a host application to manipulate and inspect data before and after interviews are presented or documents are generated. Depending on host application design, Answer Collections are sometimes also archived for later reuse.
Present Interviews	HotDocs Server generates browser-based interviews (dynamic series of input forms) that collect the answers needed to assemble a document based on a given HotDocs template. The browser-based interview encapsulates rules and dynamic behavior imposed by the template author. When the interview is completed, answers are made available to the host application.	Host applications can place HotDocs-generated interviews directly in any standards-compliant web page.
Assemble Documents	HotDocs Server takes an answer collection and a template path as input and delivers an assembled document. It can also optionally convert most assembled documents into PDF format. Results of assembly and/or conversion are delivered to the host application.	Host applications are responsible for requesting assemblies, storing assembled documents, and making those documents available to end users in whatever way meets your requirements.
Manage "Assembly Queues"	Because of their embedded business logic, HotDocs interviews can be dynamic. What begins as a single interview or document assembly can branch out into an "assembly queue": a sort of mini-workflow containing a series of one or more interviews that need to be presented to the user and/or documents that need to be generated, as dictated by the template author.	Your host application may need to be aware of how assembly queues work in order to facilitate the user experience of progressing through the work to be done.

To see a working example of a HotDocs Server host application, you can install HotDocs Sample Portal. This basic host application (including all source code) is available as a demonstration of one way in which you could write a host application. Although it incorporates all of the basic features of HotDocs Server into a functioning host application, it is not designed to be used as-is in a production environment.

You may also choose one of several out-of-the-box host applications from HotDocs, such as HotDocs Workspace, or HotDocs Template Portal. These pre-built host applications include template, answer file, and user group management, offering an easy alternative to building your own. They can also be customized to meet your needs.

--- **Note** ---
If you need assistance developing your own host application, the HotDocs consulting team can help. It has experience in developing Web sites and template sets, and integrating existing systems and applications with HotDocs Server. Contact your HotDocs sales representative for details.

What is HotDocs Developer?

Every HotDocs Server implementation begins (in some sense) with HotDocs Developer (formerly known as HotDocs Professional). This is the desktop-based software application that is used to create the templates that will run on HotDocs Server.

When creating templates for use with HotDocs Server, you can use most of the same features as you can when creating templates for use with the desktop version of HotDocs. For example, HotDocs Server templates may contain related variables grouped into separate dialogs, dialog scripts that dynamically gray or hide variables, instructions for specifying the order in which dialogs are asked, and many other features. However, some features are either not supported or work differently in HotDocs Server templates.

The following HotDocs Developer features are designed specifically for creating HotDocs Server templates:

- **Component File Properties:** Templates may be enabled for use with HotDocs Server at the **Component File Properties** dialog box. Once enabled, templates may be used with either desktop HotDocs or HotDocs Server. Also, when this property is set, HotDocs warns you if you try to use a feature that is not allowed in a HotDocs Server template.
- **Test in Browser:** The **Test in Browser** command (**Template** menu) allows template developers to view a browser-based (either JavaScript or Silverlight) interview for HotDocs Server-enabled templates. Using this feature is a way to ensure that the interview looks and works correctly before uploading the template to the Web server.
- **Publish Templates for Use With HotDocs Server:** The **Publishing Wizard** (**Tools** menu) includes an option for publishing templates for use with HotDocs Server. When this option is selected, HotDocs generates a special file used by HotDocs Server to generate browser-based interviews: the Template Manifest file. (This file will have the same name as your template, but with “.manifest.xml” appended to the file name.)
- **Upload Templates:** After publishing the templates for use with HotDocs Server, the template files, including manifests, must be uploaded to the Web server running HotDocs Server. You may do this as part of the publishing process by creating a Web page on the server to handle the uploaded files, or you can use your own uploading process to manually upload the template files after they have been published.

System Requirements

Hardware Requirements

The hardware requirements for your system vary depending upon the type and complexity of the templates assembled. HotDocs Server performance is primarily bound by disk I/O speed, which means you should ensure that the disk system is as fast as possible. Also, additional RAM improves disk I/O speed by allowing the operating system to cache more files in system memory. The basic recommended hardware configuration is as follows:

- 2 or more processors (2 GHz or faster) (Single processor machines can work well for development and lower load production environments.)
- 1GB RAM (2 GB recommended)
- SCSI or comparable storage subsystem

Software Requirements

Your system must meet the following software requirements before you can install :

- Microsoft Windows Server 2008, 2008r2, 2012 and 2012r2

Note: When deploying on Windows Server 2012, it is recommended that the temporary directory be located on an NTFS partition rather than ReFS.

- Microsoft .NET Framework 3.5 and 4.0

Note: On some operating systems, if the HotDocs Server installation program does not detect .NET Framework on your machine, the HotDocs Server installation program installs these elements automatically. On other operating systems, such as Windows Server 2008 and Windows Server 2012, you must manually install the .NET Framework before you install HotDocs Server.

- Open XML 2.5
- Microsoft Visual C++ 2008 SP1 Redistributable Package (x86)
- Microsoft Internet Information Services (IIS) 6 or greater (IIS 7 or greater is required for easy installation of Sample Portal)
 - Common HTTP Features: Static Content, Default Document.
 - Application Development: ASP.NET, .NET Extensibility.

Note: With Server 2012, you need to manually enable ASP.NET 3.5 and 4.5, as well as .NET Extensibility 3.5 and 4.5

- Security: Basic Authentication, Request Filtering.
- Management Tools: IIS Management Console, IIS Management Scripts and Tools
- Microsoft Silverlight 5 SDK (included in the HotDocs Server install)

Software Requirements for Your End Users

HotDocs server exists to generate documents from templates created in HotDocs Developer. You create a host application to present those templates to your end users in the form of interviews. Your end users use a web browser to provide the data needed to generate a finished document from the interview, and they must then be able to open the finished document.

Your end users must have certain software installed on their computers to complete browser-based interviews or view assembled documents. You should make your end users aware of their software requirements:

- A Web browser capable of displaying an interview. Examples include Microsoft Internet Explorer 8 or later, or Firefox, Chrome, or Safari. JavaScript interviews are also compatible with mobile Safari on iPads. Silverlight interviews require a desktop browser with the Silverlight 5 runtime engine installed.
- Any word processor (such as WordPerfect® or Microsoft® Word) that can display, edit, or print an assembled text document.
- HotDocs® Filler (Windows only) for assembled HFD and HPD form documents.
- Any PDF viewer (such as Adobe® Reader®) for viewing or printing assembled documents in the PDF format

Port Requirements

HotDocs Server requires certain open ports, which facilitate the remote management of HotDocs Server, as well as clustered server support:

- Port 9851: Required for all instances of HotDocs Server

HotDocs Server is dependent on Windows Management Instrumentation (WMI), which means you should add some parameters to your firewall rule group.

For Windows Server 2008, you can configure the firewall by opening Windows Firewall with Advanced Security and then going to the Inbound Rules node. From there, you should enable the following rules:

- Windows Management Instrumentation (Async-In)
- Windows Management Instrumentation (DCOM-In)
- Windows Management Instrumentation (WMI-In)

Tip: For more information on these firewall requirements, see <http://msdn2.microsoft.com/en-us/library/bb736288.aspx>.